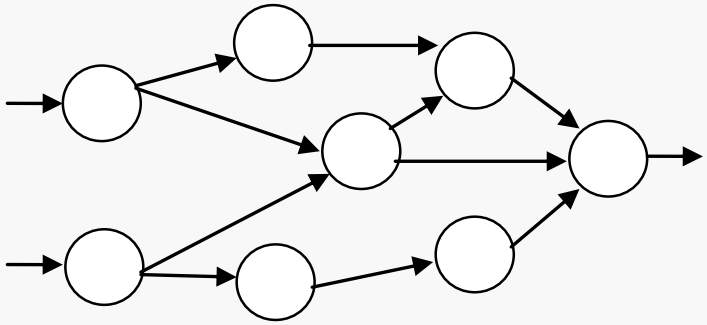
Το Πολυεπίπεδο Perceptron

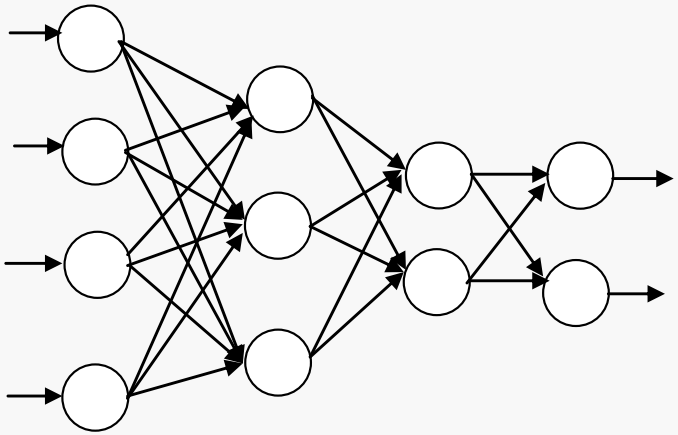
Δίκτυα Πρόσθιας Τροφοδότησης (feedforward)

- Το αντίστοιχο γράφημα του δικτύου δεν περιλαμβάνει κύκλους: **δεν υπάρχει δηλαδή ανατροφοδότηση** της εξόδου ενός νευρώνα προς τους νευρώνες από τους οποίους επηρεάζεται άμεσα ή έμμεσα.
- Οι υπολογισμοί είναι όλοι στην ίδια κατεύθυνση: από την είσοδο προς την έξοδο.
- Υλοποιούν στατική απεικόνιση από των χώρο των εισόδων R^d στο χώρο των εξόδων R^p (d είσοδοι, p έξοδοι).



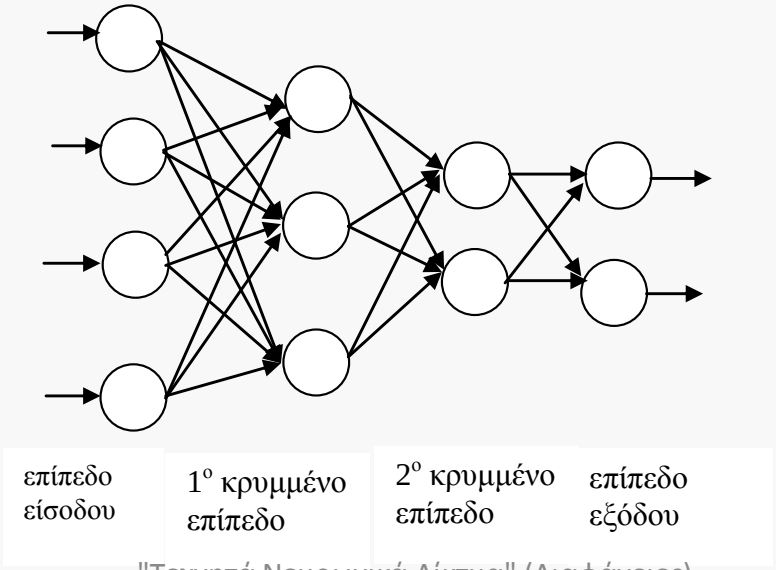
Το Πολυεπίπεδο Perceptron (MultiLayer Perceptron (MLP))

- Δίκτυο πρόσθιας τροφοδότησης (feedforward).
- Οι νευρώνες οργανωμένοι σε **επίπεδα ή στρώματα (layers):** δεν υπάρχουν συνδέσεις μεταξύ νευρώνων του ίδιου επιπέδου.
- Διαστρωμάτωση: διευκολύνει τη μαθηματική ανάλυση και επιπλέον προσφέρει **δυνατότητα παράλληλης επεξεργασίας**



Το Πολυεπίπεδο Perceptron (MLP)

- Επίπεδο εισόδου, **ένα ή περισσότερα κρυμμένα επίπεδα μη γραμμικών νευρώνων** εσωτερικού γινομένου, επίπεδο εξόδου.
- Πλήρης διασύνδεση μεταξύ των νευρώνων δύο διαδοχικών επιπέδων. Συνήθως **δεν** επιτρέπονται συνδέσεις μεταξύ νευρώνων που ανήκουν σε επίπεδα που δεν είναι διαδοχικά.



Το Πολυεπίπεδο Perceptron (MLP)

- Συμβολισμός i^{ℓ} : νευρώνας i του ℓ επιπέδου.
- $\mathrm{u}_{\mathrm{i}}^{(\ell)}$: τη συνολική είσοδο στο νευρώνα
- $y_{\mathrm{i}}^{(\ell)}$: την έξοδο του νευρώνα
- $\delta_{\mathrm{i}}^{(\ell)}$: το σφάλμα του νευρώνα
- $w_{\mathrm{i}0}^{(\ell)}$: την πόλωση του νευρώνα (ή $b_{\mathrm{i}}^{(\ell)}$)
- g_{ℓ} : τη συνάρτηση ενεργοποίησης των νευρώνων στο επίπεδο ℓ
- d_{ℓ} : τον αριθμό των νευρώνων στο επίπεδο ℓ
- $w_{\mathrm{ij}}^{(\ell)}$: βάρος της σύνδεσης από το νευρώνα $j^{\ell-1}$ στο νευρώνα i^{ℓ}

Το Πολυεπίπεδο Perceptron (MLP)

- Έστω ένα MLP με **d** εισόδους, **p** εξόδους και **H** κρυμμένα επίπεδα. Το επίπεδο εισόδου είναι το μηδέν το επίπεδο εξόδου το H+1. ($d_0 = d$, $d_{H+1} = p$)
- **Ευθύ πέρασμα (forward pass)** (δοθέντος του διανύσματος εισόδου υπολογίζεται το διάνυσμα εξόδου):
- Επίπεδο εισόδου: $y_i^{(0)} = x_i \ , y_0^{(0)} = x_0 = 1$
- Κρυμμένα επίπεδα και επίπεδο εξόδου: Για h=1,…,H+1

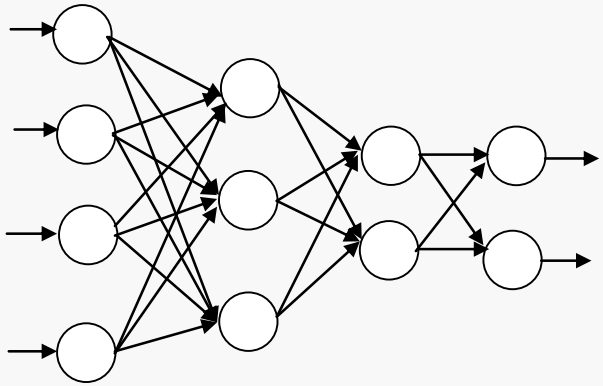
$$u_i^{(h)} = \sum_{j=0}^{d_{h-1}} w_{ij}^{(h)} y_j^{(h-1)} \ <==> \ u_i^{(h)} = \sum_{j=1}^{d_{h-1}} w_{ij}^{(h)} y_j^{(h-1)} + w_{i0}^{(h)}, \quad i=1,\ldots d_h$$

$$y_i^{(h)} = g_h(u_i^{(h)}) \quad i=1,\ldots,d_h, \quad y_0^{(h)} = 1$$

- Έξοδος του δικτύου: $o_i = y_i^{(H+1)} \qquad i=1,\ldots,p$

Το Πολυεπίπεδο Perceptron (MLP)

- Η συνάρτηση ενεργοποίησης των **κρυμμένων νευρώνων είναι μη γραμμική (συνήθως λογιστική: σ(u)=1/(1+exp(-u))).**
- Στο **επίπεδο εξόδου** η συνάρτηση ενεργοποίησης είναι συνήθως **γραμμική ή λογιστική** ανάλογα με το πρόβλημα που έχουμε να επιλύσουμε.
- Για προβλήματα ταξινόμησης προτιμάται η λογιστική και για προβλήματα συναρτησιακής προσέγγισης η γραμμική.



Υπολογιστικές Δυνατότητες του MLP

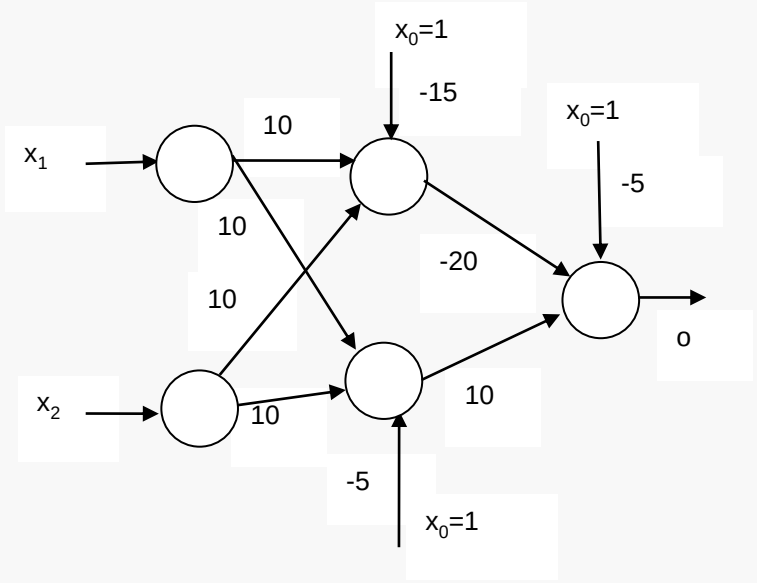
- Το MLP υλοποιεί συναρτησιακή προσέγγιση (απεικόνιση) από το χώρο των εισόδων στο χώρο των εξόδων.
- Η απεικόνιση που επιθυμούμε να υλοποιήσουμε καθορίζεται από τα παραδείγματα εκπαίδευσης.
- Το MLP χαρακτηρίζεται από την ιδιότητα της **παγκόσμιας προσέγγισης** (universal approximation): **ένα MLP** με τουλάχιστον ένα κρυμμένο επίπεδο με **μη γραμμικούς νευρώνες μπορεί να προσεγγίσει οποιαδήποτε συνάρτηση με οποιαδήποτε ακρίβεια**, αυξάνοντας επαρκώς τον αριθμό των κρυμμένων νευρώνων.
- Η ιδιότητα αυτή είναι θεωρητικά μόνο σημαντική, αλλά δεν είναι πρακτικά χρήσιμη.
- Το πρόβλημα του καθορισμού του αριθμού των κρυμμένων νευρώνων για ένα δεδομένο σύνολο εκπαίδευσης αποτελεί σήμερα βασικό ερευνητικό ζήτημα: **πρόβλημα επιλογής μοντέλου** (**model selection**).

Υπολογιστικές Δυνατότητες του MLP

- Η ύπαρξη των **μη γραμμικών κρυμμένων νευρώνων** προσδίδει στο MLP τις αυξημένες υπολογιστικές δυνατότητες.
- Το MLP **μπορεί** να επιλύσει προβλήματα ταξινόμησης που είναι **μη γραμμικά** διαχωρίσιμα.
- **Θεωρητικά** μπορεί να υλοποιήσει οποιαδήποτε επιφάνεια διαχωρισμού όσο πολύπλοκη και εάν είναι.
- Ενα MLP με σιγμοειδή συνάρτηση ενεργοποίησης των νευρώνων ορίζει υπερεπίπεδα στο χώρο των δεδομένων και δημιουργεί περιοχές απόφασης που αντιστοιχούν σε **τομές υπερεπιπέδων.**
- Συνήθως 1 ή 2 κρυμμένα επίπεδα – τα τελευταία χρόνια χρησιμοποιούνται και περισσότερα (deep neural networks)

Ασκηση

- Δίνεται το ΤΝΔ του Σχήματος του οποίου τόσο οι κρυμμένοι νευρώνες όσο και οι νευρώνες εξόδου έχουν λογιστική. Να δείξετε ότι επιλύει το πρόβλημα XOR. Για διευκόλυνση θεωρείστε ότι: i) σ(z)=0 αν z<-4, ii) σ(z)=1 αν z>4



Εκπαίδευση του MLP

- Έστω **σύνολο εκπαίδευσης D={(x^n,t^n)}**, n=1,…,N.
- $x^n=(x_{n1,\ldots},x_{nd})^T$, $t^n=(t_{n1,\ldots},t_{np})^T$ (πρόβλημα συναρτησιακής προσέγγισης).
- Θα πρέπει το MLP να έχει d νευρώνες στο επίπεδο εισόδου και p νευρώνες στο επίπεδο εξόδου.
- Θα πρέπει ο χρήστης να καθοριστεί η υπόλοιπη αρχιτεκτονική: κρυμμένα επίπεδα, αριθμός κρυμμένων νευρώνων ανά επίπεδο, είδος συναρτήσεων ενεργοποίησης.
- o(x^n; w): το διάνυσμα εξόδου του MLP όταν το διάνυσμα εισόδου είναι το x^n, και $\mathbf{w=(w_1,w_2,\ldots,w_L)^T}$ είναι ένα διάνυσμα στο οποίο συγκεντρώνουμε όλα τα βάρη και τις πολώσεις.
- Εκπαίδευση: καθορισμός του διανύσματος **w**.

Εκπαίδευση του MLP

- Στην περίπτωση που για κάποιο διάνυσμα βαρών w η εκπαίδευση **είναι τέλεια** θα ισχύει ότι (διανυσματική ισότητα):
  - **$o(x^n; w)=t^n$ για κάθε n=1,…,N**
- ή ισοδύναμα
  - $o_m(x^n; w)=t_{nm}$ για κάθε n=1,…,N, m=1,…,p
- Σε αναλογία με τον απλό νευρώνα ορίζουμε την **τετραγωνική συνάρτηση σφάλματος**

$$E(w)=\frac{1}{2}\sum_{n=1}^{N}\|t^n-o(x^n;w)\|^2=\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{p}(t_{nm}-o_m(x^n;w))^2$$

$$\updownarrow$$

$$E(w)=\sum_{n=1}^{N}E^n(w),\quad E^n(w)=\frac{1}{2}\|t^n-o(x^n;w)\|^2=\frac{1}{2}\sum_{m=1}^{p}(t_{nm}-o_m(x^n;w))^2$$

Εκπαίδευση του MLP

$$E(w)=\sum_{n=1}^{N} E^n(w), \quad E^n(w)=\frac{1}{2}\|t^n-o(x^n;w)\|^2=\frac{1}{2}\sum_{m=1}^{p}(t_{nm}-o_m(x^n;w))^2$$

- Το E(w) ως άθροισμα τετραγώνων των σφαλμάτων ανά παράδειγμα (x^n, t^n) έχει κάτω φράγμα την τιμή μηδέν η οποία προκύπτει όταν έχουμε τέλεια εκπαίδευση.
- Εκπαίδευση του MLP: **ενημέρωση του διανύσματος των βαρών w με σκοπό την ελαχιστοποίηση του τετραγωνικού σφάλματος E(w).**
- Όπως και στον απλό νευρώνα η μέθοδος ελαχιστοποίησης που έχει ευρύτερα χρησιμοποιηθεί είναι η **μέθοδος της gradient descent**.
- Χρειάζεται ο υπολογισμός των **μερικών παραγώγων** του σφάλματος E^n ως προς τα βάρη w_i :

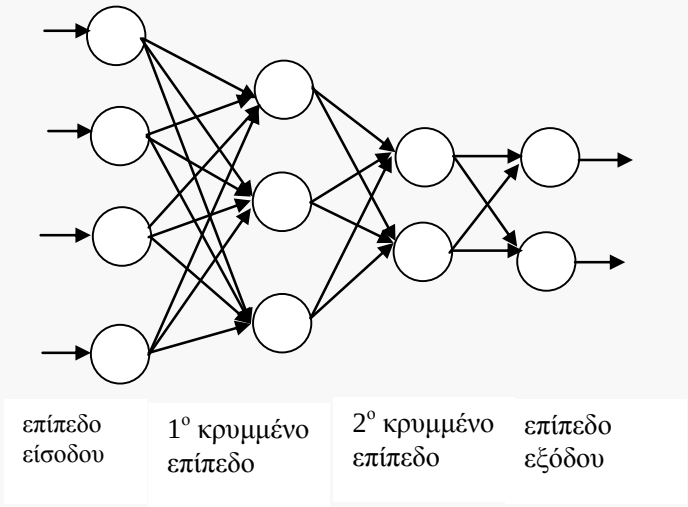
κανόνας οπισθοδιάδοσης σφάλματος (error backpropagation)

Μέθοδος backpropagation

- Τεχνική υπολογισμού των μερικών παραγώγων του σφάλματος για ένα παράδειγμα (x,t) ως προς τα βάρη σε ένα δίκτυο πρόσθιας τροφοδότησης με νευρώνες εσωτερικού γινομένου και παραγωγίσιμες συναρτήσεις ενεργοποίησης (MLP).
- Πήρε το όνομά της από το γεγονός ότι βασίζεται στην **προς τα πίσω διάδοση διαμέσου του δικτύου των σφαλμάτων που προκύπτουν στις εξόδους του δικτύου.**
- Για τον υπολογισμό των σφαλμάτων η ροή των υπολογισμών **είναι από την έξοδο προς την είσοδο.**
- **Υπολογίζονται επιμέρους τιμές σφαλμάτων για τους κρυμμένους νευρώνες του δικτύου.**

Μέθοδος backpropagation

- Έστω το παράδειγμα (x^n, t^n) και θέλουμε να υπολογίσουμε τις μερικές παραγώγους του σφάλματος E^n ως προς τα βάρη του MLP.
- Αλγόριθμος back-propagation: δύο περάσματα κατά την εκτέλεση των υπολογισμών: **ευθύ πέρασμα (forward pass)**, και το **αντίστροφο πέρασμα (reverse pass)**.



Μέθοδος backpropagation

- **Ευθύ πέρασμα**: Για διάνυσμα εισόδου x^n υπολογίζεται η έξοδος y κάθε νευρώνα του δικτύου.
- **Αντίστροφο πέρασμα (υπολογισμός σφάλματος δ κάθε νευρώνα)**
  - αρχίζει από το επίπεδο εξόδου (H+1), όπου συγκρίνονται οι τελικές έξοδοι o_i του δικτύου με τις επιθυμητές t_{ni} παράγοντας **το σφάλμα στις εξόδους** του MLP.
  - Κατόπιν **τα σήματα σφάλματος διαδίδονται προς τα πίσω στο δίκτυο** και υπολογίζεται σταδιακά **το σφάλμα για τους νευρώνες κάθε επιπέδου** από το τελευταίο κρυμμένο επίπεδο έως το πρώτο κρυμμένο επίπεδο.
- **Μερική παράγωγος βάρους σύνδεσης:**

  σφάλμα προορισμού x έξοδος πηγής

Μέθοδος backpropagation

- **Υπολογισμός σφαλμάτων (αντίστροφο πέρασμα)**
  - Νευρώνες εξόδου (επίπεδο H+1) (συν. ενεργοποίησης g_{H+1})

$$\delta_i^{(H+1)} = g'_{H+1}(u_i^{(H+1)})(o_i - t_{ni}),\ i=1,...,p$$

$$\delta_i^{(H+1)} = (o_i - t_{ni}),\ i=1,...,p\ (\text{γραμμική συν. ενεργοποίησης})$$

$$\delta_i^{(H+1)} = o_i(1-o_i)(o_i - t_{ni}),\ i=1,...,p\ (\text{λογιστική συν. ενεργοποίησης})$$

  - Νευρώνες κρυμμένων επιπέδων: για επίπεδο h=H,…,1 (συν. ενεργοποίησης g_h)

$$\delta_i^{(h)} = g'_h(u_i^{(h)})\sum_{j=1}^{d_{h+1}} w_{ji}^{(h+1)}\delta_j^{(h+1)},\ \ i=1,...,d_h$$

$$\delta_i^{(h)} = y_i^{(h)}(1-y_i^{(h)})\sum_{j=1}^{d_{h+1}} w_{ji}^{(h+1)}\delta_j^{(h+1)},\ \ i=1,...,d_h\ (\text{λογιστική συν. ενεργοποίησης})$$

Μέθοδος backpropagation

- Μερική παράγωγος βάρους σύνδεσης:

  σφάλμα προορισμού x έξοδος πηγής

$$\frac{\partial \mathrm{E}^{\mathrm{n}}}{\partial \mathrm{w}_{\mathrm{ij}}^{(\mathrm{h})}} = \delta_{\mathrm{i}}^{(\mathrm{h})} \mathrm{y}_{\mathrm{j}}^{(\mathrm{h}-1)}$$

- Μερική παράγωγος πόλωσης = σφάλμα του νευρώνα

$$\frac{\partial \mathrm{E}^{\mathrm{n}}}{\partial \mathrm{w}_{\mathrm{i0}}^{(\mathrm{h})}} = \delta_{\mathrm{i}}^{(\mathrm{h})}$$

Εκπαίδευση MLP με gradient descent (ομαδική ενημέρωση)

1. Αρχικοποίηση: Θέτουμε t:=0, αρχικοποίηση βαρών w(0) (τυχαίες τιμές στο διάστημα (-1,1)) και ρυθμού μάθησης η.
2. Σε κάθε επανάληψη t (εποχή), έστω w(t) το διάνυσμα των βαρών
   - Αρχικοποιούμε: $\frac{\partial \mathrm{E}}{\partial \mathrm{w_i}} = 0,\ \mathrm{i}=1,\ldots,\mathrm{L}$
   - Για n=1,…,N
     - εφαρμογή του κανόνα backpropagation για το (x^n,t^n) και υπολογισμός των $\frac{\partial \mathrm{E}^n}{\partial \mathrm{w_i}},\ \mathrm{i}=1,\ldots,\mathrm{L}$
     - ενημέρωση του μερικού αθροίσματος: $\frac{\partial \mathrm{E}}{\partial \mathrm{w_i}} := \frac{\partial \mathrm{E}}{\partial \mathrm{w_i}} + \frac{\partial \mathrm{E}^n}{\partial \mathrm{w_i}}$
   - Ενημέρωση των βαρών: $\mathrm{w_i}(t+1)=\mathrm{w_i}(t)-\eta\frac{\partial \mathrm{E}}{\partial \mathrm{w_i}},\ \mathrm{i}=1,..,\mathrm{L}$
   - Έλεγχος τερματισμού. Αν όχι, t:=t+1, goto 2

Εκπαίδευση MLP με gradient descent (σειριακή ενημέρωση)

1. Θέτουμε t:=0, αρχικοποίηση βαρών w(0) (τυχαίες τιμές στο διάστημα (-1,1)) και ρυθμού μάθησης n. Αρχικοποίηση του μετρητή επαναλήψεων (τ:=0) και του μετρητή εποχών (t:=0).
2. Στην αρχή κάθε επανάληψης t (εποχή), έστω w(τ) το διάνυσμα των βαρών
   - Έναρξη εποχής t, αποθήκευση του τρέχοντος διανύσματος βαρών $w_{old}=w(\tau)$. Για n=1,…,N
     - εφαρμογή του κανόνα backpropagation για το (x^n,t^n) και υπολογισμός των $\frac{\partial E^n}{\partial w_i},\ i=1,...,L$
     - Ενημέρωση των βαρών: $w_i(\tau+1)=w_i(\tau)-n\frac{\partial E^n}{\partial w_i},\ i=1,..,L$
     - τ:=τ+1
   - Τέλος εποχής t, έλεγχος τερματισμού. Αν όχι, t:=t+1, goto 2

Εκπαίδευση MLP με gradient descent

- Κριτήρια τερματισμού της εκπαίδευσης (έλεγχος στο τέλος κάθε εποχής)
  - Μικρή διαφορά στην τιμή του διανύσματος βαρών μεταξύ δύο εποχών.
  - **Μικρή διαφορά στην τιμή του ολικού σφάλματος E(w) μεταξύ δύο εποχών.**
  - Μείωση της τιμής του ολικού σφάλματος E(w) κάτω από μια επιθυμητή τιμή.
  - **Πρόωρο σταμάτημα** (early stopping): χρήση συνόλου επικύρωσης.

MLP για προβλήματα ταξινόμησης (κωδικοποίηση κατηγοριών)

- **Κωδικοποίηση των κατηγοριών**: μετατροπή του προβλήματος ταξινόμησης σε πρόβλημα συναρτησιακής προσέγγισης, μέσω της αντιστοίχισης κάθε κατηγορίας σε κάποιο διάνυσμα (ή τιμή) εξόδου.
- Το αρχικό σύνολο εκπαίδευσης με ζεύγη της μορφής **(δεδομένο, κατηγορία)** μετασχηματίζεται ώστε να περιέχει ζεύγη της μορφής **(δεδομένο, διάνυσμα στόχος)**.

- **Κωδικοποίηση 1-από-p (1-out of-p)** για πρόβλημα p κατηγοριών $C_1,\ldots,C_p$
  - Κάθε διάνυσμα-στόχος έχει p συνιστώσες $(t_1,\ldots,t_p)$ και η κατηγορία C_k κωδικοποιείται θέτοντας $t_k=1$ και $t_i=0$ για $i\neq k$.

MLP για προβλήματα ταξινόμησης (κωδικοποίηση κατηγοριών)

- Παράδειγμα: σε ένα πρόβλημα με τρεις κατηγορίες, τα αντίστοιχα τρία διανύσματα εξόδου είναι τα (1,0,0), (0,1,0), (0,0,1)
  - Απαιτείται ένα MLP με τρεις εξόδους.
- Η ταξινόμηση ενός προτύπου γίνεται εφαρμόζοντας το πρότυπο ως είσοδο στο δίκτυο και **επιλέγοντας την κατηγορία που αντιστοιχεί στην έξοδο με τη μεγαλύτερη τιμή**.
  - Όσο πιο κοντά στο 1 είναι αυτή η έξοδος και κοντά στο μηδέν οι υπόλοιπες έξοδοι, τόσο πιο αξιόπιστη είναι η ταξινόμηση.

MLP για προβλήματα ταξινόμησης (κωδικοποίηση κατηγοριών)

- Ειδικά **για την περίπτωση δύο κατηγοριών**, μπορεί να χρησιμοποιηθεί και κωδικοποίηση με μία έξοδο:
  - αντιστοιχίζουμε την έξοδο t=1 στην μια κατηγορία (C_1)
  - και την έξοδο t=0 στην άλλη κατηγορία (C_2).
- Στην περίπτωση αυτή, η ταξινόμηση ενός δεδομένου εισόδου γίνεται ως εξής:
  - αν η έξοδος είναι μεγαλύτερη του 0.5 τότε το πρότυπο ταξινομείται στην κατηγορία C_1, αλλιώς στην κατηγορία C_2.